**Practica Calificada 1 – Sección 1**

Profesor Heider Sanchez

**Alumno:**

**Instrucciones:**

* La prueba dura 110 minutos.
* Resuelva los ejercicios en un documento diferente.
* No copie los enunciados, solo lo necesario para resolver el ejercicio.
* Mantenga el orden de los enunciados en el documento de resolución.
* Queda prohibido cualquier comunicación entre alumnos.
* **Está prohibido revisar cualquier página web que no sea Canvas.**
* **Sino sabe la respuesta, deje la pregunta en blanco.**

1. **(8 pts) Responder las preguntas en el Kahoot**
2. **(2 pts) Registros de Longitud Fija: Eliminación**

Dada la siguiente tabla de datos, se le aplicar la eliminación de registros usando la técnica de: mover el ultimo registro de la tabla a la posición del registro a eliminar.

“data.dat”

|  |  |  |  |
| --- | --- | --- | --- |
| 7 registros | |  |  |
|  | Id | Nombre | Ciclo |
| 1 | P-102 | Andrea | 5 |
| 2 | P-150 | Carlos | 7 |
| 3 | P-362 | Cinthya | 3 |
| 4 | P-931 | Josimar | 5 |
| 5 | P-187 | Jorge | 1 |
| 6 | P-312 | Mabel | 3 |
| 7 | P-082 | Keiko | 9 |

1. (1 pts) Muestre el estado de la tabla después de eliminar:

P-082, P-102, P-362, P-150 (en ese orden)

|  |  |  |  |
| --- | --- | --- | --- |
| 6 registros | | Delete P-082 |  |
|  | Id | Nombre | Ciclo |
| 1 | P-102 | Andrea | 5 |
| 2 | P-150 | Carlos | 7 |
| 3 | P-362 | Cinthya | 3 |
| 4 | P-931 | Josimar | 5 |
| 5 | P-187 | Jorge | 1 |
| 6 | P-312 | Mabel | 3 |
| 7 | P-082 | Keiko | 9 |

|  |  |  |  |
| --- | --- | --- | --- |
| 5 registros | | Delete P-102 |  |
|  | Id | Nombre | Ciclo |
| 1 | P-312 | Mabel | 3 |
| 2 | P-150 | Carlos | 7 |
| 3 | P-362 | Cinthya | 3 |
| 4 | P-931 | Josimar | 5 |
| 5 | P-187 | Jorge | 1 |
| 6 | P-102 | Andrea | 5 |
| 7 | P-082 | Keiko | 9 |

|  |  |  |  |
| --- | --- | --- | --- |
| 4 registros | | Delete P-362 |  |
|  | Id | Nombre | Ciclo |
| 1 | P-312 | Mabel | 3 |
| 2 | P-150 | Carlos | 7 |
| 3 | P-187 | Jorge | 1 |
| 4 | P-931 | Josimar | 5 |
| 5 | P-362 | Cinthya | 3 |
| 6 | P-102 | Andrea | 5 |
| 7 | P-082 | Keiko | 9 |

|  |  |  |  |
| --- | --- | --- | --- |
| 4 registros | | Delete P-150 |  |
|  | Id | Nombre | Ciclo |
| 1 | P-312 | Mabel | 3 |
| 2 | P-362 | Cinthya | 3 |
| 3 | P-187 | Jorge | 1 |
| 4 | P-931 | Josimar | 5 |
| 5 | P-150 | Carlos | 7 |
| 6 | P-102 | Andrea | 5 |
| 7 | P-082 | Keiko | 9 |

1. (1 pts) Muestre el estado de la tabla después de insertar:

|  |  |  |
| --- | --- | --- |
| P-390 | Peter Castle | 10 |

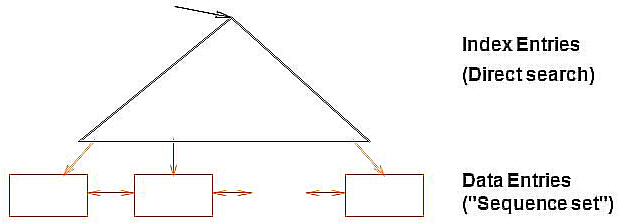
|  |  |  |
| --- | --- | --- |
| P-011 | Lee Yoo-mi | 1 |

|  |  |  |  |
| --- | --- | --- | --- |
| 4 registros | | Insert P-390 |  |
|  | Id | Nombre | Ciclo |
| 1 | P-312 | Mabel | 3 |
| 2 | P-362 | Cinthya | 3 |
| 3 | P-187 | Jorge | 1 |
| 4 | P-931 | Josimar | 5 |
| 5 | P-390 | Peter Castle | 10 |
| 6 | P-102 | Andrea | 5 |
| 7 | P-082 | Keiko | 9 |

|  |  |  |  |
| --- | --- | --- | --- |
| 4 registros | | Insert P-011 |  |
|  | Id | Nombre | Ciclo |
| 1 | P-312 | Mabel | 3 |
| 2 | P-362 | Cinthya | 3 |
| 3 | P-187 | Jorge | 1 |
| 4 | P-931 | Josimar | 5 |
| 5 | P-390 | Peter Castle | 10 |
| 6 | P-011 | Lee Yoo-mi | 1 |
| 7 | P-082 | Keiko | 9 |

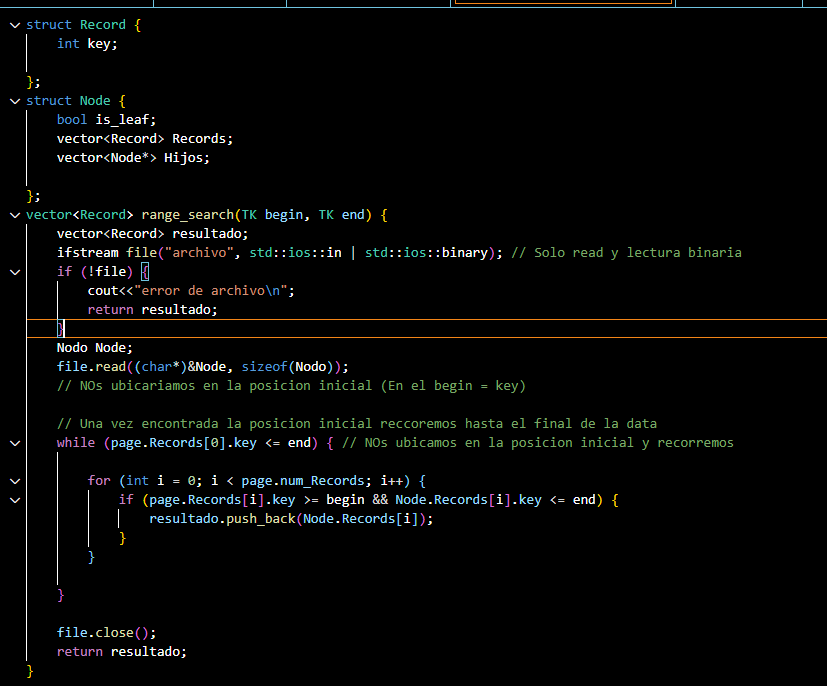
1. **(4 pts) B+ Tree File (agrupado)**

El B+Tree File es una estrategia de organización de archivos dinámica, es decir, el índice crece y decrece para ajustarse a la cantidad exacta de elementos. En el B+ Tree agrupado, cada entrada de las hojas apunta a una pagina de datos.



1. (1 pts) Diseñe la estructura de datos de la página de índice y pagina de datos.
2. (3 pts) Diseñe el algoritmo de búsqueda por rango para el índice B+ Tree (clustered), en memoria secundaria lo más optimo posible. Asuma que tenemos un archivo de datos gigantesco y el rango de búsqueda es muy pequeño.

vector<Record> range\_search(TK begin, TK end)



1. **(3 pts) Dynamic Hashing**

Construya el índice hash dinámico (árbol digital) a partir de los siguientes datos:

**­­**

Tome como profundidad global 𝐷=3 y un factor de bloque 𝑓𝑏=3. **Los bits se van tomando de izquierda a derecha.**

1

0

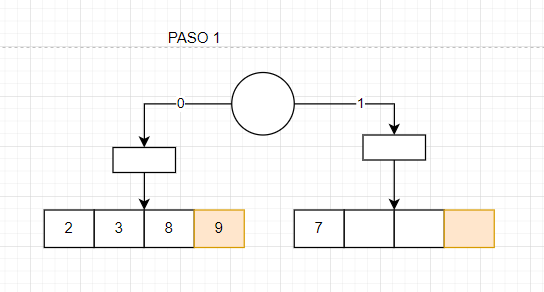
1. (1 pts) Muestre la tabla de keys y códigos binarios

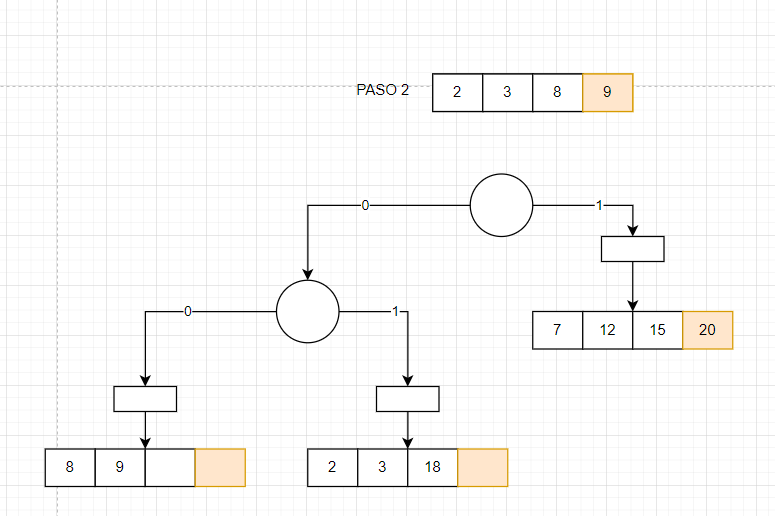
Tabla Hash

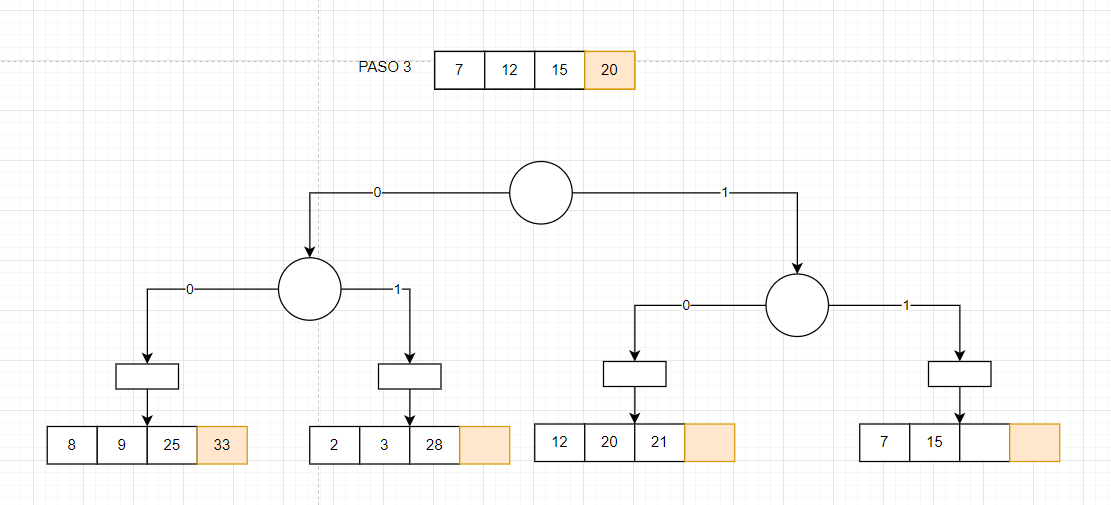
|  |  |  |
| --- | --- | --- |
| key | HASH | BIN |
| 2 | 2 | 010 |
| 3 | 3 | 011 |
| 7 | 7 | 111 |
| 8 | 0 | 000 |
| 9 | 1 | 001 |
| 12 | 4 | 100 |
| 15 | 7 | 111 |
| 18 | 2 | 010 |
| 20 | 4 | 100 |
| 21 | 5 | 101 |
| 25 | 1 | 001 |
| 33 | 1 | 001 |
| 38 | 6 | 110 |
| 47 | 7 | 111 |
| 54 | 6 | 110 |
| 61 | 5 | 101 |
| 65 | 1 | 001 |

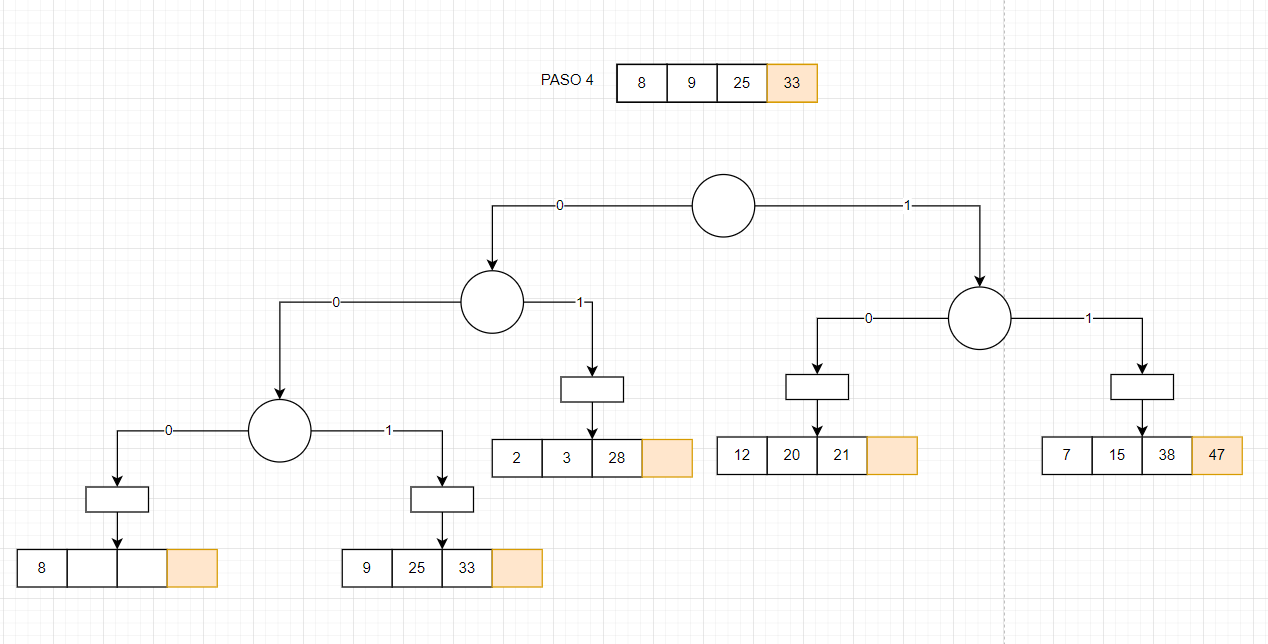
1. (2 pts) ¿Cómo queda el índice? Muestre el árbol final y los buckets.

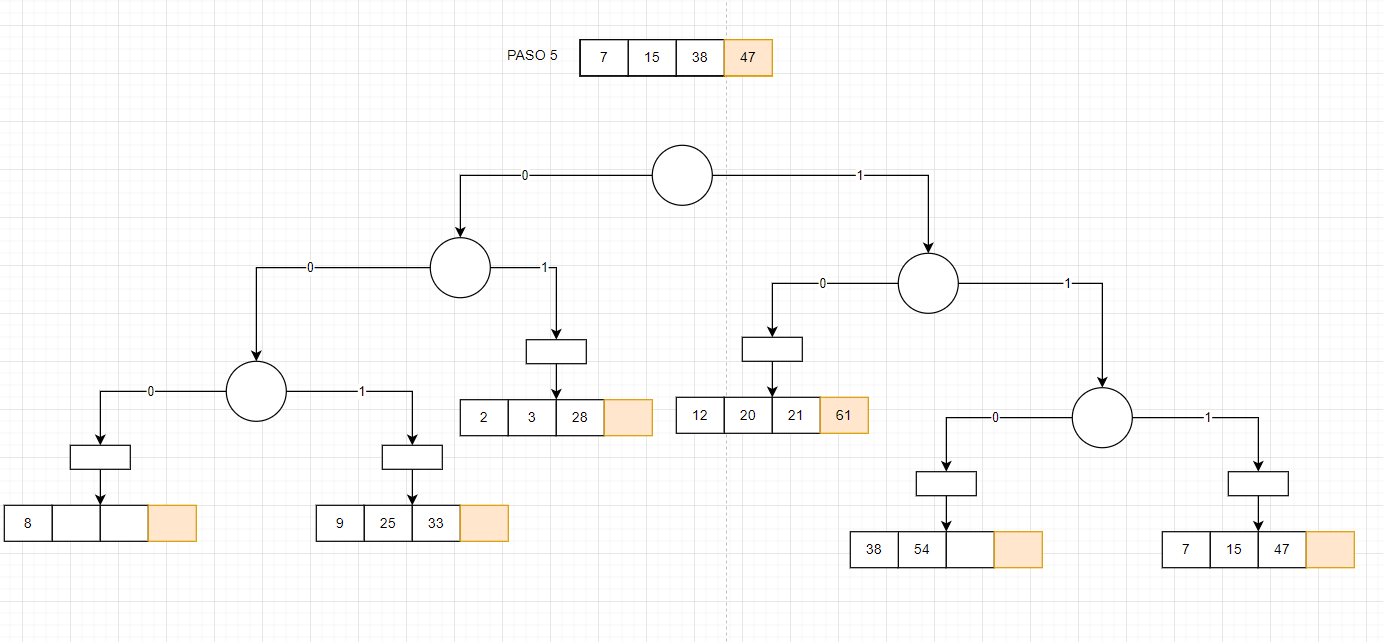
Inicio inserciones árbol (izq -> der)

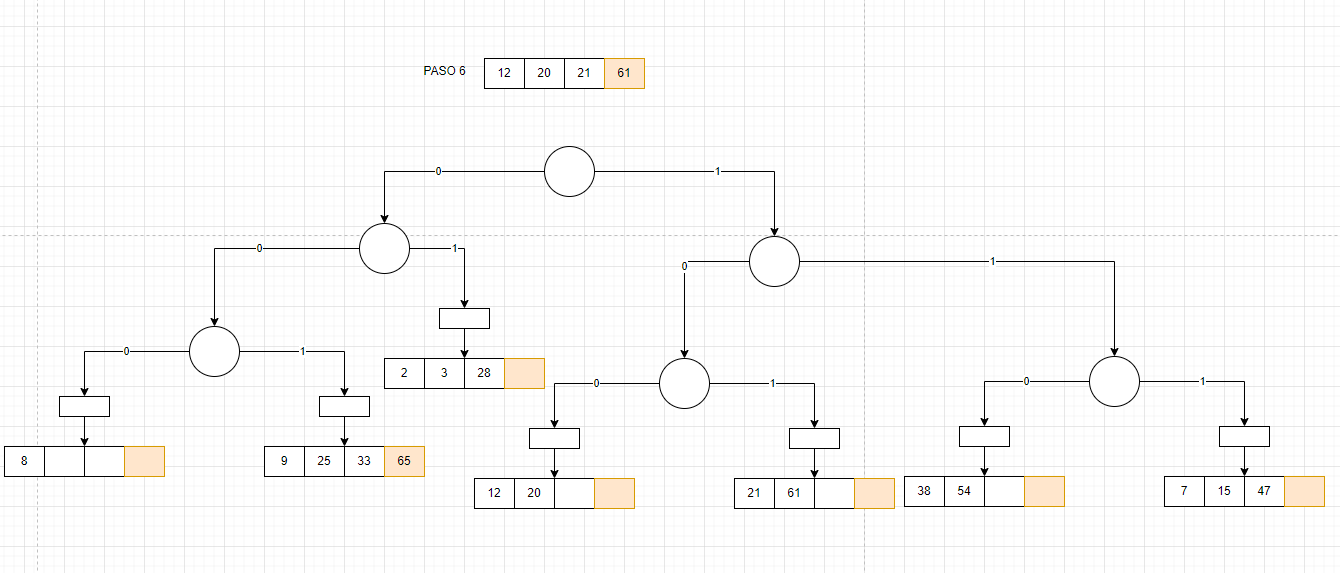


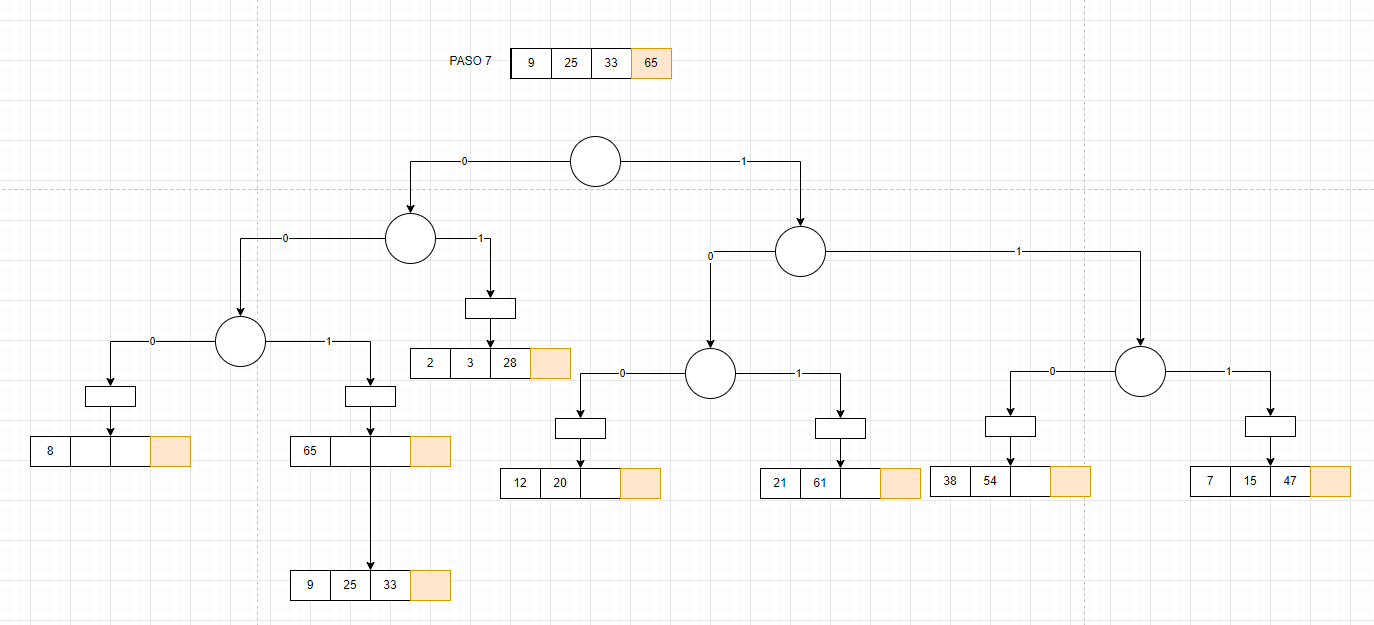




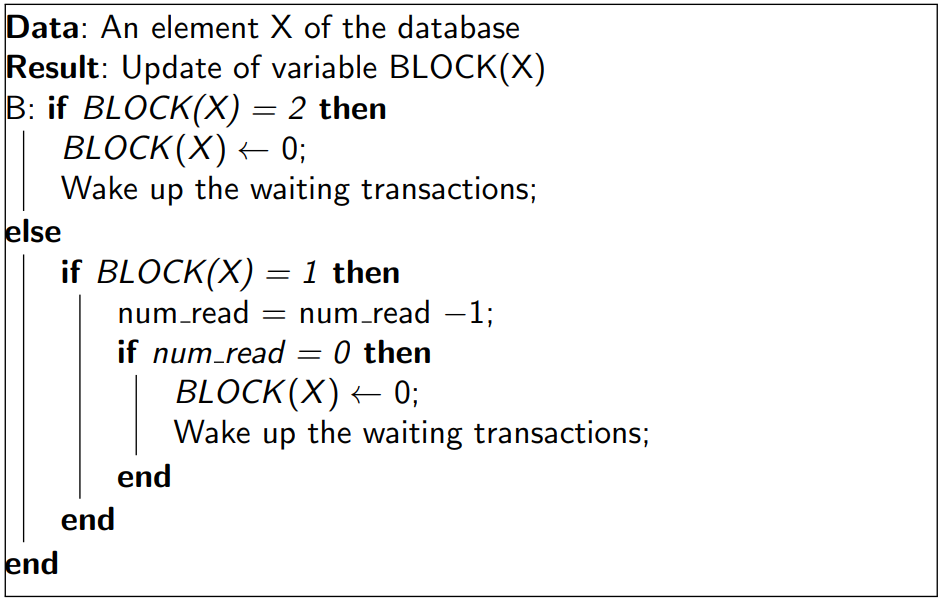


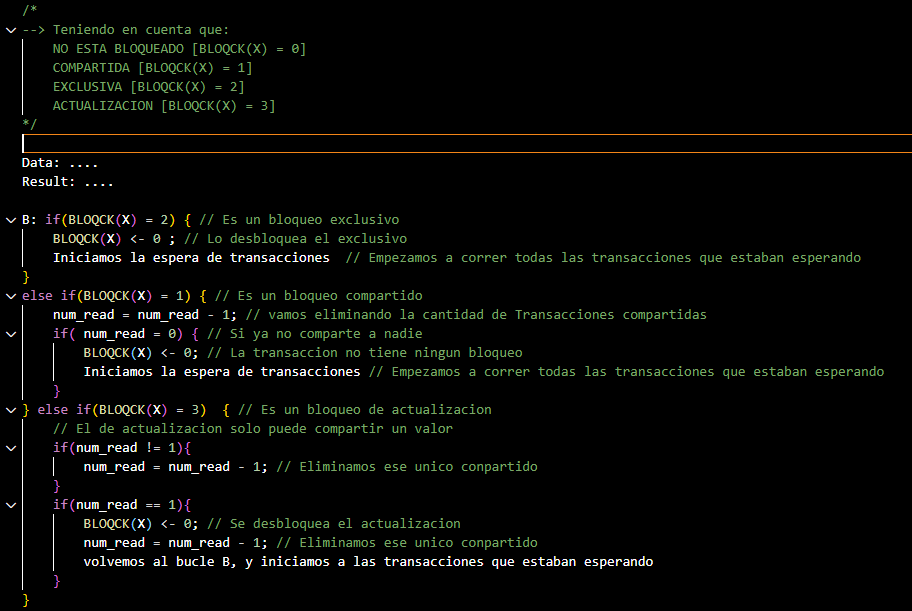






1. **(3.5 pts) Control de Concurrencia**
2. (1.5 pts) ¿Cómo modificaría el algoritmo de desbloqueo para considerar también el caso cuando el recurso X está bloqueado con protocolo PU?.

****



1. (2 pts) Database recovery

Dado el siguiente registro del log de transacciones, se le pide ejecutar el algoritmo de recuperación basado en Checkpoint y mostrar los valores de los recursos de la BD.

<T1, BT>

<T1, A, 10, 15>

<T2, BT>

<T2, B, 20, 10>

<T2, C, 11, 12>

<T1, D, 17, 18>

<T2, COMMIT>

<CHECKPOINT>

<T1, C, 12, 20>

<T3, BT>

<T3, A, 15, 12>

<T3, B, 20, 12>

<T4, BT>

<T4, C, 20, 22>

<T3, COMMIT>

Falla del sistema

A=10, B=20, C=11, D=17   
A=15, B=20, C=11, D=17   
A=15, B=10, C=11, D=17  
A=15, B=10, C=12, D=17  
A=15, B=10, C=12, D=18

A=15, B=10, C=20, D=18 A=12, B=20, C=12, D=18  
  
A=12, B=12, C=20, D=18

A=12, B=20, C=20, D=18

A=12, B=12, C=22, D=18

* Valores de los recursos antes de ejecutar la planificación

BD: A=10, B=20, C=11, D=17

* Valores de los recursos antes de la falla del sistema

BD: A=12, B=12, C=22, D=18

* Valores de los recursos después de la recuperación del sistema

BD: A=12, B=20, C=12, D=18